UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/889,350	07/17/2001	Martin Russ	36-1466	2630

23117 7590 06/13/2007

NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
BLACKWELL, JAMES H	
ART UNIT	PAPER NUMBER
2176	

MAIL DATE	DELIVERY MODE
06/13/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 09/889,350	Applicant(s) RUSS ET AL.	
	Examiner James H. Blackwell	Art Unit 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *26 January 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17,20,21,23-27 and 29-47* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) *1-17,20,21,23-27 and 29-47* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. In view of the Appeal Brief filed on 01/26/2007, PROSECUTION IS HEREBY REOPENED. A new set of rejections is set forth below.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) initiate a new appeal by filing a notice of appeal under 37 CFR 41.31 followed by an appeal brief under 37 CFR 41.37. The previously paid notice of appeal fee and appeal brief fee can be applied to the new appeal. If, however, the appeal fees set forth in 37 CFR 41.20 have been increased since they were previously paid, then appellant must pay the difference between the increased fees and the amount previously paid.

2. A Supervisory Patent Examiner (SPE) has approved of reopening prosecution by signing below.

3. Claims 1-17, 20-21, 23-27, and 29-47 remain pending in this application.

4. Claims 1, 23-24, 29, and 46-47 are independent claims.

5. The original priority date of the application was **02/17/1999**.

Allowable Subject Matter

6. Claims 8 and 36 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1-2, 4, 17, 20-21, 23-24, 29-30, 32, and 45-47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Windows 95 ("Introducing Microsoft Windows 95," Copyright 1981-1995, Microsoft, Pages 22, 38, 40) in view of Christiansen (pod2html documentation, copyright 05/15/1997, 14 pages total, downloaded from <http://annocpan.org/~CHIPS/perl5.004/*.*>)

In regard to independent Claim 1, <u>Windows 95</u> discloses:

- *A method of creating hypermedia content for a web site, making use of a computer configuration that has an operating system wherein files are stored in folders arranged in a hierarchical file structure provided by a file handling system native to the operating system, and the operating system is provided with a viewer that provides a visual display of the hierarchical file structure and an indication of the file content* (Pg. 22, Figures on page;→ Windows Explorer, a component of the Windows 95 Operating System for creating, viewing, and otherwise manipulating any number or type of folders, subfolders, and files within a hierarchical structure and for visualizing the hierarchical structure), *the method comprising:*

- *using the viewer of the operating system to view the hierarchical file structure corresponding to content for the web site* (Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner), *and*

Windows 95 fails to disclose:

- *running a conversion program module to convert the file contents stored in folders arranged in the hierarchical file structure provided by the file handling system native to the operating system into hypermedia for the web site with hyperlinks therein corresponding to at least one relationship of at least one of the stored files with at least one other of the stored files within the hierarchical file structure provided by the file handling system native to the operating system.*

However, Christiansen discloses *a conversion program module to convert the file contents stored in folders arranged in the hierarchical file structure provided by the file handling system native to the operating system into hypermedia for the web site with hyperlinks therein corresponding to at least one relationship of at least one* *of the stored files with at least one other of the stored files within the hierarchical file structure provided by the file handling system native to the operating system* (Pgs. 11-14;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 11 through Pg. 14 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to dependent Claim 2, Windows 95 discloses:

- *using the viewer to transfer files of content for the web site from other file locations of the computer configuration, into the hierarchical file structure for the web site* (Pg. 38, files can be moved from one folder to the other by dragging and dropping, see Figures on page).

In regard to dependent Claim 4, Windows 95 discloses:

- *the hierarchical file structure includes a first one of the folders for storing files, and at least one underlying layer containing at least one file sub-folder* (see Figs. Pgs. 22, 38, 40;→ a file viewing system with directories and sub-directories for containing files).

Windows 95 fails to disclose:

- *the conversion program module produces web pages corresponding to the first of the folders and any sub-folders with hyperlinks between them corresponding to folder hierarchy.*

However, Christiansen discloses *a conversion program module produces web pages corresponding to the first of the folders and any sub-folders with hyperlinks between them corresponding to folder hierarchy* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 1 through Pg. 4 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to dependent Claim 17, Windows 95 fails to disclose:

- *causing the conversion program module to be downloaded to the computer configuration from a remote server.*

However, Christiansen discloses *causing the conversion program module to be downloaded to the computer configuration from a remote server* (Pgs. 1-10;→ the

Perl script pod2html is a component of a larger Perl distribution. Typically, these distributions are archived on remote distribution sites (e.g., CPAN website, <"http://www.cpan.org/">) with the intention of providing interested parties access to the distribution for installation on their client machine, or another machine).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, and Christiansen and because both inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Christiansen provides for ready access to conversion programs.

In regard to dependent Claim 20, Windows 95 fails to disclose:

- *recorded on a recording medium insertable into the computer configuration to be loaded therein.*

However, Christiansen discloses *recorded on a recording medium insertable into the computer configuration to be loaded therein* (Pgs. 1-10;→ the Perl script pod2html is a component of a larger Perl distribution. Typically, these distributions are archived on remote distribution sites (e.g., CPAN website, <"http://www.cpan.org/">) or can be obtained on distributable media (e.g., CD-ROM) with the intention of providing interested parties access to the distribution for installation on their client machine, or another machine).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, and Christiansen and because both inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Christiansen provides for ready access to conversion programs.

In regard to dependent Claim 21, Windows 95 fails to disclose:

- *configured as a download to be downloaded from a server to the computer configuration.*

However, Christiansen discloses *configured as a download to be downloaded from a server to the computer configuration* (Pgs. 1-10;→ the Perl script pod2html is a component of a larger Perl distribution. Typically, these distributions are archived on remote distribution sites (e.g., CPAN website, <"http://www.cpan.org/">) or can be obtained on distributable media (e.g., CD-ROM) with the intention of providing interested parties access to the distribution for installation on their client machine, or another machine).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, and Christiansen and because both inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Christiansen provides for ready access to conversion programs.

In regard to independent Claim 23, Windows 95 discloses:

- *A method of creating hypermedia content for a web site, the method comprising:*
    - *storing files in folders arranged in a hierarchical file structure provided by a file handling system native to an operating system; providing a visual display of the hierarchical file structure and an indication of file content with a viewer that is provided with the operating system* (Pg. 22, Figures on page;→ Windows Explorer, a component of the Windows 95 Operating System for creating, viewing, and otherwise manipulating any number or type of folders, subfolders, and files within a hierarchical structure and for visualizing the hierarchical structure).
    - *using the hierarchical file structure provided by the native handling system of the operating system to define at least one relationship between at least one of the files and at least one other of the files* (Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner).

Windows 95 fails to disclose:

- *converting the file content into hypermedia for the web site with hyperlinks therein corresponding the at least one relationship between the at least one of the files and at the least one of the other files as defined by the hierarchical file structure through execution of a conversion program module.*

However, Christiansen discloses *converting the file content into hypermedia for the web site with hyperlinks therein corresponding the at least one relationship between the at least one of the files and at the least one of the other files as defined by the hierarchical file structure through execution of a conversion program module* (Pgs. 11-14;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 11 through Pg. 14 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--

htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to independent Claim 24, Windows 95 discloses:

- *A computer system for creating hypermedia content for a web site, the computer system comprising:*
    - *an operating system for configuring and storing files in folders arranged in a hierarchical structure, the operating system being provided with a native file handling system defining the hierarchical structure and a viewer that provides a visual display of the hierarchical structure and an indication of file content* (Pg. 22, Figures on page;→ Windows Explorer, a component of the Windows 95 Operating System for creating, viewing, and otherwise manipulating any number or type of folders, subfolders, and files within a hierarchical structure and for visualizing the hierarchical structure).

Windows 95 fails to disclose:

- *a conversion program module for converting the hierarchical structure corresponding to content for the web site previously assembled in the operating system using the viewer, into hypermedia for the web site with hyperlinks therein based on one or more identified relationships between folders storing files, the one or more identified relationships being defined by an arrangement of the folders within the hierarchical structure.*

However, Christiansen discloses *a conversion program module for converting the hierarchical structure corresponding to content for the web site previously assembled in the operating system using the viewer, into hypermedia for the web site with hyperlinks therein based on one or more identified relationships between folders storing files, the one or more identified relationships being defined by an arrangement of the folders within the hierarchical structure* (Pgs. 11-14;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 11 through Pg. 14 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---

podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to independent Claim 29, Windows 95 discloses:

- *A method of creating hypermedia content for a web-site from content files stored in a memory of a computer system,*
    - *the computer system having an operating system which includes a file management function, with which each file may be stored within a folder or a sub-folder and each folder or sub-folder may contain zero, one or more sub-folders to enable a hierarchical file structure to be formed in which files in a first folder are considered to be higher up the hierarchical*

file structure than files stored within a sub-folder of the first folder (Pg. 22, Figures on page;→ Windows Explorer, a component of the Windows 95 Operating System for creating, viewing, and otherwise manipulating any number or type of folders, subfolders, and files within a hierarchical structure and for visualizing the hierarchical structure), *and*

- ○ *a viewer which provides a visual display of the hierarchical structure and an indication of the file content* (Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner), *the method comprising:*

<u>Windows 95</u> fails to disclose:

- *traversing at least a part of the hierarchical file structure to identify one or more of the content files and its or their relationships with at least one other of the content files*

However, <u>Christiansen</u> discloses *traversing at least a part of the hierarchical file structure to identify one or more of the content files and its or their relationships with at least one other of the content files* (Pgs. 11-14;→ pod2html arguments can be appended to the basic command-line invocation (see beginning Pg. 11 through Pg. 14 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format).

Windows 95 further fails to disclose:

- *converting the one or more identified content files into hypermedia content, the converting step including generating hyperlinks within the hypermedia content which correspond to the identified relationship or relationships with the at least one other of the content files.*

However, Christiansen discloses *converting the one or more identified content files into hypermedia content, the converting step including generating hyperlinks within the hypermedia content which correspond to the identified relationship or relationships with the at least one other of the content files* (Pgs. 11-14, ARGUMENTS Section;→ pod2html allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to dependent Claim 30, Windows 95 discloses:

- *using the viewer to transfer files of content for the web site from other file locations of the computer configuration, into the hierarchical file structure for the web site* (Pg. 38, files can be moved from one folder to the other by dragging and dropping, see Figures on page).

In regard to dependent Claim 32, Windows 95 discloses:

- *the hierarchical file structure includes a first one of the folders for storing files, and at least one underlying layer containing at least one file sub-folder* (see Figs. Pgs. 22, 38, 40;→ a file viewing system with directories and sub-directories for containing files).

Windows 95 fails to disclose:

- *the conversion program module produces web pages corresponding to the first of the folders and any sub-folders with hyperlinks between them corresponding to folder hierarchy.*

However, Christiansen discloses *a conversion program module produces web pages corresponding to the first of the folders and any sub-folders with hyperlinks between them corresponding to folder hierarchy* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to

Unix “manpage” text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 1 through Pg. 4 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95 and Christiansen because both Windows Explorer and pod2html operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through a user interface. Adding the disclosure of Christiansen provides a tool that can be invoked by the user to convert files in an existing file structure from their present form into hypermedia files usable on the World Wide Web.

In regard to dependent Claim 45, Windows 95 fails to disclose:

- *causing the conversion program module to be downloaded to the computer configuration from a remote server.*

However, Christiansen discloses *causing the conversion program module to be downloaded to the computer configuration from a remote server* (Pgs. 1-10;→ the Perl script pod2html is a component of a larger Perl distribution. Typically, these distributions are archived on remote distribution sites (e.g., CPAN website, <"http://www.cpan.org/">) with the intention of providing interested parties access to the distribution for installation on their client machine, or another machine).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, and Christiansen and because both inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Christiansen provides for ready access to conversion programs.

In regard to Claim 46, Claim 46 merely recites a computer system for performing the method of Claim 29. Thus, the combination of Windows 95 with Christiansen discloses every limitation of Claim 46, as indicated in the above rejection for Claim 29.

In regard to Claim 47, Claim 47 merely recites a computer readable medium tangibly embodying a computer program for executing the method of Claim 29. Thus, the combination of Windows 95 with Christiansen discloses every limitation of Claim 47, as indicated in the above rejection for Claim 29.

9. Claims 3, 5-7, 9-13, 31, 33-35, and 37-41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Windows 95 in view of Christiansen, and in further view of Drakos ("The LaTeX2HTML Translator", 02/22/1995, Univ. of Leeds, 36 pages).

In regard to dependent Claim 3, Windows 95 fails to disclose:

- *... using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site.*

However, Christiansen discloses *... using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file.

Pod files are of a single "format" and typically do not contain other types of content (e.g., graphics, audio, video). Thus, Christiansen fails to expressly disclose *... using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site.*

However, Drakos discloses *using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site* (Pg. 14, Example; Pg. 15, Fig. 1;→ a Perl script to convert LaTeX and other formats of files associated with a LaTeX document into equivalent HTML. LaTeX2HTML can convert files (4th line of LaTeX snippet, figure.ps) with different formats (e.g. PostScript) into a form suitable for use as hypermedia (e.g., Fig. 1 shows how figure.ps is rendered by the converter). As listed in the LaTeX snippet, the PostScript file is, by virtue of the path listed (actually lack thereof) to be, by default, in the same directory as the LaTeX file. Thus, the given directory has both a LaTeX source file (text, ASCII) and another format of file (PostScript).

Therefore, Drakos discloses a converter that can convert multiple formats of files into hypermedia

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Both Christiansen and Drakos further perform conversion of files from an initial (non-hypermedia) format to HTML (a hypermedia format). Adding the disclosure of Drakos further provides for the conversion of multiple file types into hypermedia usable on the World Wide Web.

In regard to dependent Claim 5, both Windows 95 and Christiansen fail to disclose:

- *the first folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the first folder, and the conversion program module produces a web page corresponding to a template for the node.*

However, Drakos discloses *the first folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the first folder, and the conversion program module produces a web page corresponding to a template for the node* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web.

In regard to dependent Claim 6, both Windows_95 and Christiansen fail to disclose:

- *the conversion program module produces a web page corresponding to the template for a node corresponding to the at least one sub-folder.*

However, Drakos discloses *the conversion program module produces a web page corresponding to the template for a node corresponding to the at least one sub-folder* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It is noted that this limitation is being interpreted as simply converting a template file in a sub-directory, which is the same as converting a template file in any directory. ,

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 7, Windows 95 fails to disclose:

- *the conversion program module searches the at least one sub-folder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the at least one sub-folder and the produces a web page corresponding to the template for a node corresponding to the at least one sub-folder.*

However, Christiansen discloses *the conversion program module searches the at least one sub-folder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the at least one sub-folder and the produces a web page corresponding to the template for a node corresponding to the at least one sub-folder* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old

Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 1 through Pg. 4 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

Christiansen fails to expressly disclose *templates.*

However, Drakos discloses *templates* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 9, both Windows 95 and Christiansen fail to disclose:

- *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template.*

However, Drakos discloses *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template* (Pg. 11, Sec. 3.6; snippet; → the snippet allows for the addition of HTML-specific content, such as the addition of URL's as well as the importation of images (.gif) from other locations).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 10, both Windows 95 and Christiansen fail to disclose:

- *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link.*

However, Drakos discloses *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link* (Pg. 11, Sec. 3.6; snippet; → the snippet allows for the addition of HTML-specific content, such as the addition of URL's as well as the importation of images (.gif) from other locations).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 11, both Windows 95 and Christiansen fail to disclose:

- *the first folder or the at least one sub-folder contains a document template for defining a predetermined configuration for hypermedia at a web page in the web site corresponding to a text document in the first folder or the at least one sub-folder.*

However, Drakos discloses *the first folder or the at least one sub-folder contains a document template for defining a predetermined configuration for hypermedia at a web page in the web site corresponding to a text document in the first folder or the at least one sub-folder* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It is noted that this limitation is being interpreted as simply converting a template file in a directory; that is, any directory.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 12, Windows 95 discloses:

- *providing a library of said templates and using the file viewer to transfer a selected one or more of the templates from the library to the one or more of the folders* (Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files (including template files) are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner).

  It is noted that the viewer depicted in Windows 95 would have been capable of both viewing and moving files (including template files) from one location in the directory structure to another, and that a given directory could have contained multiple files, file types (including templates) from which a user could pick and choose.

In regard to dependent Claim 13, Windows 95 discloses:

- *the folder structure hierarchy includes a root folder and sub-folders depending therefrom, and including placing at least one of the templates in the root folder* Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files (including template files) are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner).

It is noted that the viewer depicted in Windows 95 would have been capable of both viewing and moving files (including template files) from one location in the directory structure to another, and that a given directory could have contained multiple files, file types (including templates) from which a user could pick and choose.

In regard to dependent Claim 31, Windows 95 fails to disclose:

- *... using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site.*

However, Christiansen discloses ... *using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file.

Pod files are of a single "format" and typically do not contain other types of content (e.g., graphics, audio, video). Thus, Christiansen fails to expressly disclose ... *using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web site.*

However, Drakos discloses (Pg. 14, Example; Pg. 15, Fig. 1;→ a Perl script to convert LaTeX and other formats of files associated with a LaTeX document into equivalent HTML. LaTeX2HTML can convert files (4th line of LaTeX snippet, figure.ps) with different formats (e.g. PostScript) into a form suitable for use as hypermedia (e.g., Fig. 1 shows how figure.ps is rendered by the converter). As listed in the LaTeX snippet, the PostScript file is, by virtue of the path listed (actually lack thereof) to be, by default, in the same directory as the LaTeX file. Thus, the given directory has both a LaTeX source file (text, ASCII) and another format of file (PostScript).

Therefore, Drakos discloses a converter that can convert multiple formats of files into hypermedia

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Both Christiansen and Drakos further perform conversion of files from an initial (non-hypermedia) format to HTML (a hypermedia format). Adding the disclosure of Drakos further provides for the conversion of multiple file types into hypermedia usable on the World Wide Web.

In regard to dependent Claim 33, both Windows 95 and Christiansen fail to disclose:

- *the first folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the first folder, and the conversion program module produces a web page corresponding to a template for the node.*

However, Drakos discloses *the first folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the first folder, and the conversion program module produces a web page corresponding to a template for the node* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web.

In regard to dependent Claim 34, both Windows 95 and Christiansen fail to disclose:

- *the conversion program module produces a web page corresponding to the template for a node corresponding to the at least one sub-folder.*

However, Drakos discloses *the conversion program module produces a web page corresponding to the template for a node corresponding to the at least one sub-folder* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It is noted that this limitation is being interpreted as simply converting a template file in a sub-directory, which is the same as converting a template file in any directory.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 35, Windows 95 fails to disclose:

- *the conversion program module searches the at least one sub-folder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the at least one sub-folder and the produces a web page corresponding to the template for a node corresponding to the at least one sub-folder.*

However, Christiansen discloses *the conversion program module searches the at least one sub-folder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding* *to the at least one sub-folder and the produces a web page corresponding to the template for a node corresponding to the at least one sub-folder* (Pgs. 1-4;→ pod2html is a Perl script which allows a user to convert Pod files ("Plain Old Documentation", similar to Unix "manpage" text files containing documentation content and some markup). The script perl2html is called from a command-line interface and takes, in its simplest form, as input a Pod formatted file and outputs the equivalent html formatted file. Arguments can also be appended to the basic command-line invocation (see beginning Pg. 1 through Pg. 4 under heading entitled ARGUMENTS). Using these arguments, perl2html can be instructed to traverse, from a base directory (--podroot-name), following a defined path from the root directory to sub-directories (---podpath=name:...:name), and recursively process content in these sub-directories (--recurse) from Pod to HTML format. In addition, pod2html also allows for the definition of an HTML base URL for the converted Pod files (--htmlroot=name) and automatically creates hyperlinks between the HTML-converted Pod files).

Christiansen fails to expressly disclose *templates*.

However, Drakos discloses *templates* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 37, both Windows 95 and Christiansen fail to disclose:

- *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template.*

However, Drakos discloses *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template* (Pg. 11, Sec. 3.6; snippet; → the snippet allows for the addition of HTML-specific content, such as the addition of URL's as well as the importation of images (.gif) from other locations).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 38, both Windows 95 and Christiansen fail to disclose:

- *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link.*

However, Drakos discloses *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link* (Pg. 11, Sec. 3.6; snippet; → the snippet allows for the addition of HTML-specific content, such as the addition of URL's as well as the importation of images (.gif) from other locations).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 39, both Windows 95 and Christiansen fail to disclose:

- *the first folder or the at least one sub-folder contains a document template for defining a predetermined configuration for hypermedia at a web page in the web site corresponding to a text document in the first folder or the at least one sub-folder.*

However, Drakos discloses *the first folder or the at least one sub-folder contains a document template for defining a predetermined configuration for hypermedia at a web page in the web site corresponding to a text document in the first folder or the at least one sub-folder* (Abstract;→ converting LaTeX files (TeX, LaTeX languages are typically used for typesetting of content by "marking up" the content with various typesetting commands). Pg. 11, Sec. 3.6 of Drakos also discloses embedded HTML (i.e., hypermedia, a URL, see snippet) within a LaTeX document.

Thus, one of ordinary skill in the art at the time of invention would reasonably interpret such a file as a template containing (defining) a predetermined configuration for hypermedia (the LaTeX file once converted would contain a hyperlink at the section of the LaTeX file containing the LaTeX markup embedding the URL). Upon conversion, the LaTeX "marked up" content is converted to an equivalent HTML file (by default, it the same folder as the LaTeX source).

It is noted that this limitation is being interpreted as simply converting a template file in a directory; that is, any directory.

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Drakos because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces. Adding the disclosure of Drakos provides for the conversion of LaTeX files (templates) into hypermedia usable on the World Wide Web

In regard to dependent Claim 40, Windows 95 discloses:

- *providing a library of said templates and using the file viewer to transfer a selected one or more of the templates from the library to the one or more of the folders* (Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files (including template files) are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner).

It is noted that the viewer depicted in Windows 95 would have been capable of both viewing and moving files (including template files) from one location in the directory structure to another, and that a given directory could have contained multiple files, file types (including templates) from which a user could pick and choose.

In regard to dependent Claim 41, Windows 95 discloses:

- *the folder structure hierarchy includes a root folder and sub-folders depending therefrom, and including placing at least one of the templates in the root folder* Pg. 22, Figures on Page;→ Windows Explorer is used on a computer (e.g., such as that storing/hosting a web site), where various types and kinds of files (including template files) are arranged in a hierarchical series of folders, subfolders and provides a means to view the files in a hierarchical manner).

It is noted that the viewer depicted in Windows 95 would have been capable of both viewing and moving files (including template files) from one location in the directory structure to another, and that a given directory could have contained multiple files, file types (including templates) from which a user could pick and choose.

10. Claims 14-16, 25-27, and 42-44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Windows 95 in view of Christiansen, and in further view of Wishnie et al. (hereinafter Wishnie, U.S. Patent No. 6,148,311 filed 04/25/1997, issued 11/14/2000).

In regard to dependent Claim 14, both Windows 95 and Christiansen fail to disclose:

- *the computer configuration comprises a network and including arranging the files in the hierarchical file structure for the web site, using the file viewer, from different file locations in the network.*

However, Wishnie discloses *the computer configuration comprises a network and including arranging the files in the hierarchical file structure for the web site, using the file viewer, from different file locations in the network* (Col. 11, lines 53-67;→ web site constructor system that is networkable via I/O bus connected to serial link, LAN, wireless link, etc. Systems and tools for generating/creating web sites typically can access networks for many purposes including remote file maintenance (e.g., maintaining a web site on a remote server), file upload (uploading or publishing files to a web server).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 15, both Windows 95 and Christiansen fail to disclose:

- *uploading the hypermedia for installation on a server for the web site.*

However Wishnie discloses *uploading the hypermedia for installation on a server for the web site* (Col. 4, lines 5-7;→ once site content has been created, the content is uploaded to a server (remote site) so it can be accessed by others).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 16, both Windows 95 and Christiansen fail to disclose:

- *providing a local preview of the hypermedia before uploading it to the server.*

However, Wishnie discloses *providing a local preview of the hypermedia before uploading it to the server* (see Figs. 4a-b, 7;→ provides the user a view of both the structure and rendered views of content for the created web site).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 25, both Windows 95 and Christiansen fail to disclose:

- *operating system is executed in a networked environment for concurrent access by multiple users.*

However, Wishnie discloses *operating system is executed in a networked environment for concurrent access by multiple users* (Col. 11, lines 53-67;→ web site constructor system that is networkable via I/O bus connected to serial link, LAN, wireless link, etc. Systems and tools for generating/creating web sites typically can access networks for many purposes including remote file maintenance (e.g., maintaining a web site on a remote server), file upload (uploading or publishing files to a web server).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to Claim 26, Claim 26 merely recites a system for carrying out the method of Claim 25. Thus, the combination of Windows 95, Christiansen, and Wishnie discloses every limitation of Claim 26, as indicated in the above rejection for Claim 25.

In regard to dependent Claim 27, both Windows 95 and Christiansen fail to disclose:

- *operating system is executed in a networked environment for concurrent access by multiple users.*

However, Wishnie discloses *operating system is executed in a networked environment for concurrent access by multiple users* (Col. 11, lines 53-67;→ web site constructor system that is networkable via I/O bus connected to serial link, LAN, wireless link, etc. Systems and tools for generating/creating web sites typically can access networks for many purposes including remote file maintenance (e.g., maintaining a web site on a remote server), file upload (uploading or publishing files to a web server).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 42, both Windows 95 and Christiansen fail to disclose:

- *the computer configuration comprises a network and including arranging the files in the hierarchical file structure for the web site, using the file viewer, from different file locations in the network.*

However, Wishnie discloses *the computer configuration comprises a network and including arranging the files in the hierarchical file structure for the web site, using the file viewer, from different file locations in the network* (Col. 11, lines 53-67;→ web site constructor system that is networkable via I/O bus connected to serial link, LAN, wireless link, etc. Systems and tools for generating/creating web sites typically can access networks for many purposes including remote file maintenance (e.g., maintaining a web site on a remote server), file upload (uploading or publishing files to a web server).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 43, both Windows 95 and Christiansen fail to disclose:

- *uploading the hypermedia for installation on a server for the web site.*

However Wishnie discloses *uploading the hypermedia for installation on a server for the web site* (Col. 4, lines 5-7;→ once site content has been created, the content is uploaded to a server (remote site) so it can be accessed by others).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

In regard to dependent Claim 44, both Windows 95 and Christiansen fail to disclose:

- *providing a local preview of the hypermedia before uploading it to the server.*

However, Wishnie discloses *providing a local preview of the hypermedia before uploading it to the server* (see Figs. 4a-b, 7;→ provides the user a view of both the structure and rendered views of content for the created web site).

It would have been obvious to one of ordinary skill in the art at the time of invention to combine the disclosures of Windows 95, Christiansen, and Wishnie because all three inventions operate on computers as modules to perform functions on existing file structures (directories or folders, files, sub-directories, sub-folders) through user interfaces for the creation and manipulation of web or hypermedia content. Adding the disclosure of Wishnie provides for access and maintenance of web or other file content from a remote (networked) system.

Response to Arguments

11. Applicant's arguments with respect to claims 1, 12, 23, 24, 29, 40, 46, 47 (as argued in the Appeal Brief filed 01/26/2007) have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to James H. Blackwell whose telephone number is 571-272-4089. The examiner can normally be reached on Mon-Fri.

13. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather R. Herndon can be reached on 571-272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

14. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

James H. Blackwell
06/10/2007

Doug Hutton
Primary Examiner
Technology Center 2100